Chapter 3
Modeling in Time Domain

Goals

- Find a mathematical model, called a state space representation, for a linear time invariant system.
- Convert between transfer function and state-space models.
- Linearize a state-space representation.

3.1 Introduction

The classical control theory and methods (such as root locus) that we have been using in class to date are based on a simple input-output description of the plant, usually expressed as a transfer function. These methods do not use any knowledge of the interior structure of the plant, and limit us to single-input single-output (SISO) systems, and as we have seen allows only limited control of the closed-loop behavior when feedback control is used.

Modern control theory solves many of the limitations by using a much "richer" description of the plant dynamics.
The so-called state-space description provide the dynamics as a set of coupled first-order differential equations in a set of internal variables known as *state variables*, together with a set of algebraic equations that combine the state variables into physical output variables.

3.1 Introduction

State differential equations are an alternative way to describe a dynamic system (i.e. time-domain method)
Basic technology: A physical system can be described by a high order differential equation

an th - order diff. equation → a set of 1st - order diff. equations using state variables

$$y(t) = a_0 f(t) + a_1 f'(t) + a_2 f''(t) + \cdots a_n f^{(n)}(t)$$

$$\begin{aligned}
x_1'(t) &= a_{11}x_1(t) + a_{12}x_2(t) + \cdots + a_{1n}x_n(t) \\
x_2'(t) &= a_{21}x_1(t) + a_{22}x_2(t) + \cdots + a_{2n}x_n(t) \\
&\vdots \qquad\qquad\qquad\qquad \vdots \\
x_n'(t) &= a_{n1}x_1(t) + a_{n2}x_2(t) + \cdots + a_{nn}x_n(t)
\end{aligned}$$

3.1 Introduction

- Frequency domain techniques

  (classical/communication/USA).

  - Differential equations $\rightarrow$ algebraic equations
  - Only for linear time-invariant systems.
  - Simplify modeling of interconnected systems.
  - Easy stability analysis.
  - Limitations: number of variables, multi-input multi-output, non-zero initial conditions.

3.1 Introduction

- Time-domain Techniques
  (modern/mathematics/Russia)
  - State-space representation
  - Can handle nonlinear and time-varying systems
  - Can handle MIMO systems
  - Can handle non-zero initial conditions
  - Can handle complex systems with many variables
  - Numerous state-space software packages
  - It is not intuitive as classical approaches

3.1 Introduction

Definitions

• Linear combination of n variables x_i :

$$s = k_n\, x_n + k_{n-1}\, x_{n-1} + ... \qquad + k_1\, x_1$$

• Linear independence: No variable can be written as a linear combination of the others.

• System variable: Any variable that responds to an input or initial condition in a system.

• State variables: The smallest set of linearly independent system variables such that their values at time t_0 together with the values of the forcing function(s) completely specify the values of all system variables for $t > t_0$.

3.1 Introduction

Definitions

• State vector: a vector whose elements are the state variables.

• State space: an *n*-dimensional space whose axes are the state variables.

• State equations: a set of simultaneous, first order differential equations in the state variables.

• Output equations: a set of algebraic equations that express the system outputs as linear combinations of the state variables and the inputs.

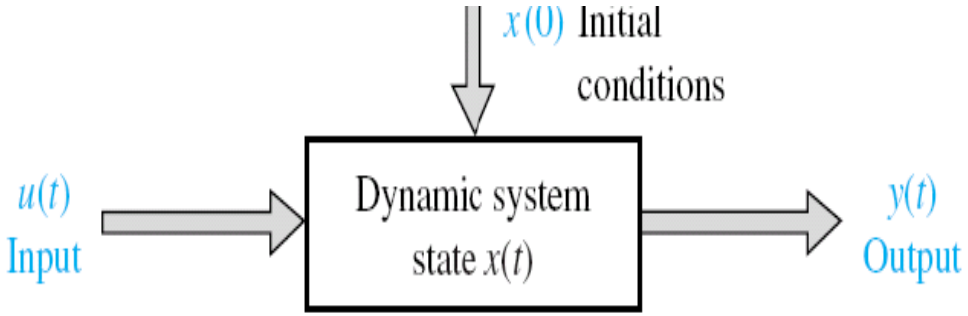
A General Dynamic System

The state of a system is described in terms of state variables

$$[x_1(t), x_2(t), x_3(t), \cdots, x_n(t)]$$

Given the input u(t), initial conditions $x_0(t)$, and the present state x(t), we can determine

- the future system behavior
- the output



3.1 Introduction

State Variables of a Dynamic System

"The state of a system is a set of variables such that the knowledge of these variables and of the input function will, with the equations describing the dynamics, provide the future state and output of the system"

"The state variables describe the future response of a system, given the present state, the excitation inputs and the equations of the system"

Russian control school: Lyapunov, Pontryagin, etc. always worked in time domain

The state is the minimum information required about a system to predict its future

3.1 Introduction

Typically for a physical system, the state relates to the concept of energy – each state describes an energy storage component (mass, spring, inductor, capacitor etc.)

Often it is useful to use variables that are readily بسرعة measured, e.g. currents, voltages, positions, velocities, pressures, temperatures, concentrations, etc…

3.2 Some Observations

Procedure

Select a subset of system variables (n state variables).

Write n first-order differential equations in terms of the n state variabl for an n^{th} order system.

Write down the output equations (algebraic equations)

Total = state-space representation

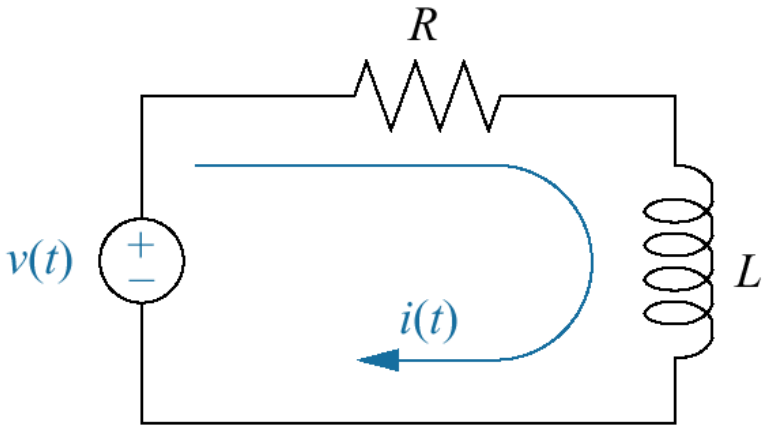
3.2 Some Observations

Example: RL-circuit

Parameters: $R,\ L$

Variables: $v(t), i(t), v_R(t), \dfrac{di(t)}{dt}$

Equation : $v(t) = Ri(t) + L\dfrac{di(t)}{d(t)}$



first order differential equation, so we need one state variable, one first-order differential equation, one initial condition.

- Suppose we want to solve for $v_R(t)$, $v_L(t)$, and $\dfrac{di(t)}{dt}$

$$v_R(t) = Ri(t)$$

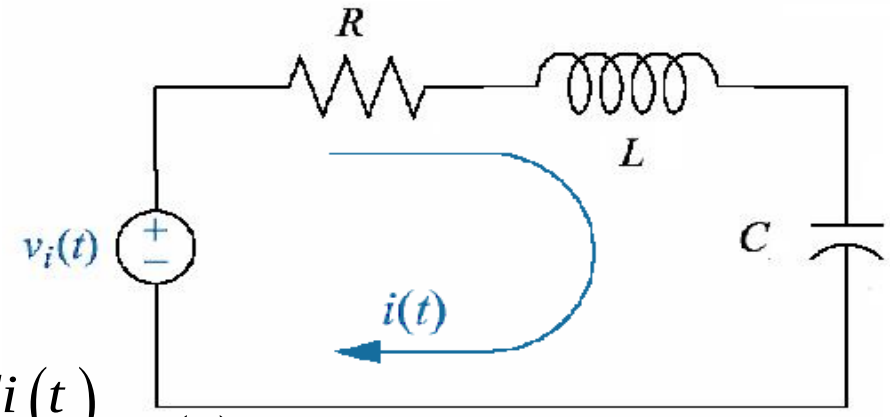
$$v_L(t) = v(t) - v_R(t) = v(t) - Ri(t)$$

$$\frac{di(t)}{dt} = \frac{v(t) - Ri(t)}{L}$$

**

3.2 Example: RLC Circuit

Example RLC-circuit



Parameters: R , L , C

Variables: $v_i(t)$, $i(t)$, $v_R(t)$, $v_C(t)$, $\frac{di(t)}{dt}$, $q(t)$

$i(t) = \frac{dq(t)}{dt}$ q : Capacitor charge

Equation : $v_i(t) = Ri(t) + L\frac{di(t)}{d(t)} + \frac{1}{C}\int i(t)dt$

So we need 2 state variables, 2 first-order differential equations, 2 initial condtions.

State variables can be selected in different pairs

let us use $i(t) = \frac{dq(t)}{d(t)}$

$$v_i(t) = R\frac{dq(t)}{dt} + L\frac{d^2q(t)}{dt^2} + \frac{1}{C}q(t) \qquad *$$

3.2 Example: RLC Circuit

can be written as two 1st order differantial equations

$$\frac{dq}{dt} = i \qquad (1)$$

$$\frac{di(t)}{dt} = -\frac{R}{L}i - \frac{1}{LC}q + \frac{1}{L}v_i(t) \qquad (2)$$

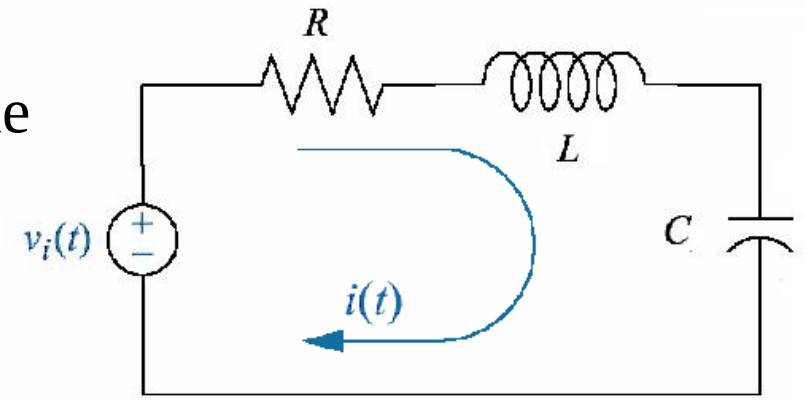
$$\begin{bmatrix} \frac{dq}{dt} \\ \frac{di(t)}{dt} \end{bmatrix} = \begin{pmatrix} 0 & 1 \\ -\frac{1}{LC} & -\frac{R}{L} \end{pmatrix} \begin{bmatrix} q(t) \\ i(t) \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{L} \end{bmatrix} v_i(t)$$

output (v_L)

$$v_L = v(t) - Ri(t) - \frac{1}{c}\int i(t)dt$$

$$= v_i(t) - Ri(t) - \frac{1}{c}q(t) \Rightarrow v_L = \begin{bmatrix} -\frac{1}{c} & -R \end{bmatrix} \begin{bmatrix} q(t) \\ i(t) \end{bmatrix} + [\,1\,]\ v_i(t)$$

3.2 Example: RLC Circuit



State-space representations are NOT unique

let the state space variable

i(t) and v_c

with $v_c = \frac{1}{c}\int i(\tau)\, d\tau \Rightarrow \frac{dv_c(t)}{dt} = \frac{1}{c} i(t)$

and

$$v_i(t) = v_R(t) + v_L(t) + v_c(t) \Rightarrow v_i(t) = R\, i(t) + L\frac{di(t)}{dt} + v_c(t)$$

$$\Rightarrow \frac{di(t)}{dt} = -\frac{R}{L} i - \frac{1}{L} v_c(t) + \frac{1}{L} v_i(t)$$

then

$$\frac{dv_c(t)}{dt} = \frac{1}{c} i(t)$$

$$\frac{di(t)}{dt} = -\frac{R}{L} i - \frac{1}{L} v_c(t) + \frac{v_i(t)}{L} \Rightarrow \begin{bmatrix} \frac{dv_c(t)}{dt} \\ \frac{di(t)}{dt} \end{bmatrix} = \begin{bmatrix} 0 & \frac{1}{c} \\ -\frac{1}{L} & -\frac{R}{L} \end{bmatrix} \begin{bmatrix} v_c(t) \\ i(t) \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{v_i(t)}{L} \end{bmatrix}$$

output (v_L)

$$v_L = v_i(t) - R\, i(t) - v_c(t) \Rightarrow v_L = v_i(t) - R\, i(t) - v_c(t) \Rightarrow v_L = \begin{bmatrix} -1 & -R \end{bmatrix} \begin{bmatrix} v_c(t) \\ i(t) \end{bmatrix} + v_i(t)$$

3.2 Example: Spring-Mass-Damper

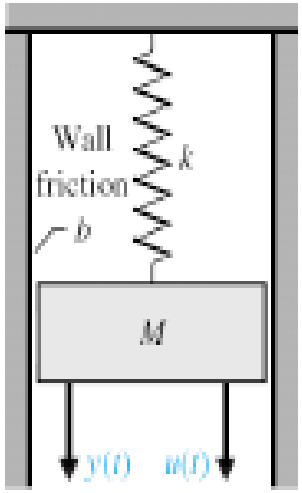
- First-order differential equations

$$\begin{aligned} \frac{dx_1(t)}{dt} &= x_2(t) \\ \frac{dx_2(t)}{dt} &= -\frac{b}{M}x_2(t) - \frac{k}{M}x_1(t) + \frac{1}{M}u(t) \end{aligned}$$

- In matrix form:

$$\begin{bmatrix} \frac{dx_1(t)}{dt} \\ \frac{dx_2(t)}{dt} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{M} & -\frac{b}{M} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{M} \end{bmatrix} u(t)$$

$$y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} x(t)$$



- with

$$A = \begin{bmatrix} 0 & 1 \\ -\frac{k}{M} & -\frac{b}{M} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ \frac{1}{M} \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad D = 0$$

3.3 the general State-Space Representation

In general

State-space representation for $t \geq t_0$, and initial conditions $\mathbf{x}(t_0)$

State equations: $\dot{\mathbf{x}} = \mathbf{Ax} + \mathbf{Bu}$

Output equations : $\mathbf{y} = \mathbf{Cx} + \mathbf{Du}$

$\mathbf{A}$ = system matrix
$\mathbf{B}$ = input matrix
$\mathbf{C}$ = output matrix
D = feedforward constant

$$\dot{\mathbf{x}} = \underbrace{\begin{bmatrix} \frac{dx_1}{dt} \\ \frac{dx_2}{dt} \\ \mathrm{M} \\ \frac{dx_n}{dt} \end{bmatrix}}_{\text{State Derivatives}} \quad \mathbf{x} = \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ \mathrm{M} \\ x_n \end{bmatrix}}_{\text{States}} \quad \mathbf{y} = \underbrace{\begin{bmatrix} y_1 \\ y_2 \\ \mathrm{M} \\ y_p \end{bmatrix}}_{\text{Output}} \quad \mathbf{u} = \underbrace{\begin{bmatrix} u_1 \\ u_2 \\ \mathrm{M} \\ u_m \end{bmatrix}}_{\text{Input}}$$

A is $(n \times n)$, B is $(n \times m$, C is $(p \times n)$ and D is $(p \times m)$. Shorthand for this system is $[A, B, C, D]$

3.3 the general State-Space Representation

Back to the example

$$x_1 = q(t) \qquad u = v(t)$$
$$x_2 = \dot{q}(t) \qquad y = v_L(t)$$

$$\dot{x}_1 = x_2$$
$$\dot{x}_2 = -\frac{1}{LC}x_1 - \frac{R}{L}x_2 + \frac{1}{L}u$$
$$y = -\frac{1}{C}x_2 - Rx_2 + u$$

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ \frac{-1}{LC} & -\frac{R}{L} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{L} \end{bmatrix} u$$

$$y = \begin{bmatrix} -\frac{1}{C} & -R \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + [1]u$$

3.3 the general State-Space Representation

Comments

- No state variable can be chosen if it can be expressed as a linear combination of the other state.

- State variables are linearly independent

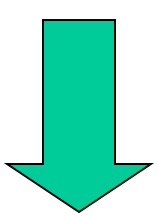
$$x_j \neq \sum_{i=1}^{n-1} a_i x_i$$

- States should give full information and not repeated information.

3.3 the general State-Space Representation

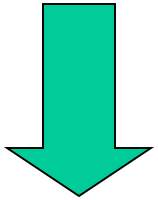
```
>> a=[0 1;-1/L*c -R/L ];
>> b=[0;1/L];
>> C=[-1/c -R];
>> d=[1];
>> df=(C*inv(s*eye(2)-a)*b+d);
>> gs1=simple(subs(df,{R L c},{1 1 1}))
```

```
>> a=[0 1/c;-1/L -R/L ];
>> b=[0;1/L];
>> C=[-1/L -R/L];
>> d=[1];
>> df=(C*inv(s*eye(2)-a)*b+d);
>> gs2=simple(subs(df,{R L c},{1 1 1}))
```



```
gs1 =

s^2/(s^2+s+1)
```



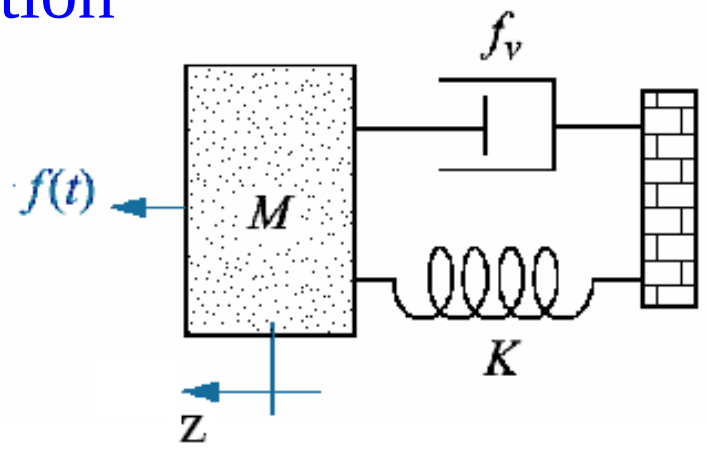
```
gs2 =

s^2/(s^2+s+1)
```

The same transfer function

3.3 the general State-Space Representation

Example: Mechanical system



$$f(t) = m\ddot{z} + f_v\dot{z} + kz$$

2^{nd} order differential equation $\rightarrow$ two initial conditions $z(0), \dot{z}(0)$, and two 1^{st} order differential equations.

Variables: $f_k,\ f_{f_v},\ z, \dot{z}, f(t)$

Select $x_1 = z$

$x_2 = \dot{z}$

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = -\frac{k}{m}x_1 - \frac{f_v}{m}x_2 + \frac{1}{m}f(t)$$

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{f_v}{m} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} [u]$$

3.3 the general State-Space Representation

Output: f_{f_v} , f_k

$$f_k = kx_1$$
$$f_{f_v} = f_v x_2$$

$$y = \begin{bmatrix} f_k \\ f_{f_v} \end{bmatrix} = \begin{bmatrix} k & 0 \\ 0 & f_v \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} [u]$$

Q:What happens if the output is &&?

3.3 the general State-Space Representation

Selection of states:

Write down the energy equation of the system:

- RLC circuit: $E = \frac{1}{2}C(v_c)^2 + \frac{1}{2}L(i_L)^2$
- Mass-spring-damper system: $E = \frac{1}{2}kx^2 + \frac{1}{2}M\dot{x}^2$

Choose the states as the variables appearing in that equation

$$\Rightarrow \; x_1 = v_c \quad x_2 = i_L$$
$$\Rightarrow \; x_1 = x \quad x_2 = \dot{x}$$

3.3 The general State-Space Representation

State-Space Description

- The state-space description leads to a system of coupled, first-order differential equations
- Reduction to first-order is a general technique to transform a high-order differential equation into a state-space model

$$y(t) = a_0 f(t) + a_1 f'(t) + a_2 f''(t) + \cdots a_n f^{(n)}(t)$$

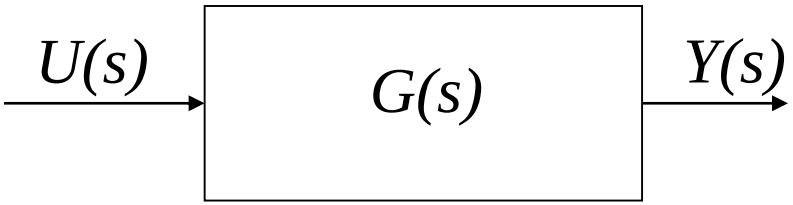
$$\begin{aligned}
x_1'(t) &= a_{11}x_1(t) + a_{12}x_2(t) + \cdots + a_{1n}x_n(t) \\
x_2'(t) &= a_{21}x_1(t) + a_{22}x_2(t) + \cdots + a_{2n}x_n(t) \\
&\vdots \qquad\qquad\qquad\qquad \vdots \\
x_n'(t) &= a_{n1}x_1(t) + a_{n2}x_2(t) + \cdots + a_{nn}x_n(t)
\end{aligned}$$

3.5 Converting a Transfer Function to State Space

$$Y(s) = G(s)U(s)$$

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u$$

$$y = \mathbf{C}\mathbf{x} + \mathbf{D}u$$



State-space representation can be better used for simulation.

Control Canonical Form

- Transfer function is unchanged when

$$\frac{Y(s)}{U(s)} = \frac{b_{n-1}s^{n-1} + \cdots + b_1 s + b_0}{s^n + a_{n-1}s^{n-1} + \cdots a_1 s + a_0} \frac{Z(s)}{Z(s)}$$

- Let

$$\begin{aligned} Y(s) &= \left(b_{n-1}s^{n-1} + \cdots + b_1 s + b_0\right) Z(s) \\ U(s) &= \left(s^n + a_{n-1}s^{n-1} + \cdots a_1 s + a_0\right) Z(s) \end{aligned}$$

- Inverse Laplace yields high-order diff. equations in z

$$\begin{aligned} y(t) &= b_{n-1}z^{(n-1)} + \cdots + b_1 \dot{z} + b_0 z \\ u(t) &= z^{(n)} + a_{n-1}z^{(n-1)} + \cdots a_1 \dot{z} + a_0 z \end{aligned}$$

State-Space Block Diagram
Control Canonical Form

- Choose the state vector

$$\begin{aligned} x_1 &= z \\ x_2 &= \dot{x}_1 \\ x_3 &= \dot{x}_2 \\ &\vdots \\ x_n &= \dot{x}_{n-1} \end{aligned}$$

- Substitute into

$$\begin{aligned} y(t) &= b_{n-1}z^{(n-1)} + \cdots + b_1\dot{z} + b_0 z \\ u(t) &= z^{(n)} + a_{n-1}z^{(n-1)} + \cdots a_1\dot{z} + a_0 z \end{aligned}$$

- To yield

$$\begin{aligned} y(t) &= b_{n-1}x_n + \cdots + b_1 x_2 + b_0 x_1 \\ u(t) &= \dot{x}_n + a_{n-1}x_n + \cdots a_1 x_2 + a_0 x_1 \end{aligned}$$

- Rearranging in state-space form,

$$\begin{aligned} \dot{x}_n &= -a_{n-1}x_n - \cdots - a_1 x_2 - a_0 x_1 + u \\ \dot{x}_{n-1} &= x_n, \quad \dot{x}_{n-2} = x_{n-1}, \quad \cdots, \quad \dot{x}_1 = x_2 \end{aligned}$$

State-Space Block Diagram
Control Canonical Form

$$\frac{Y(s)}{U(s)} = \frac{b_{n-1}s^{n-1} + \cdots + b_1 s + b_0}{s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0} \quad \text{(Standard form)}$$

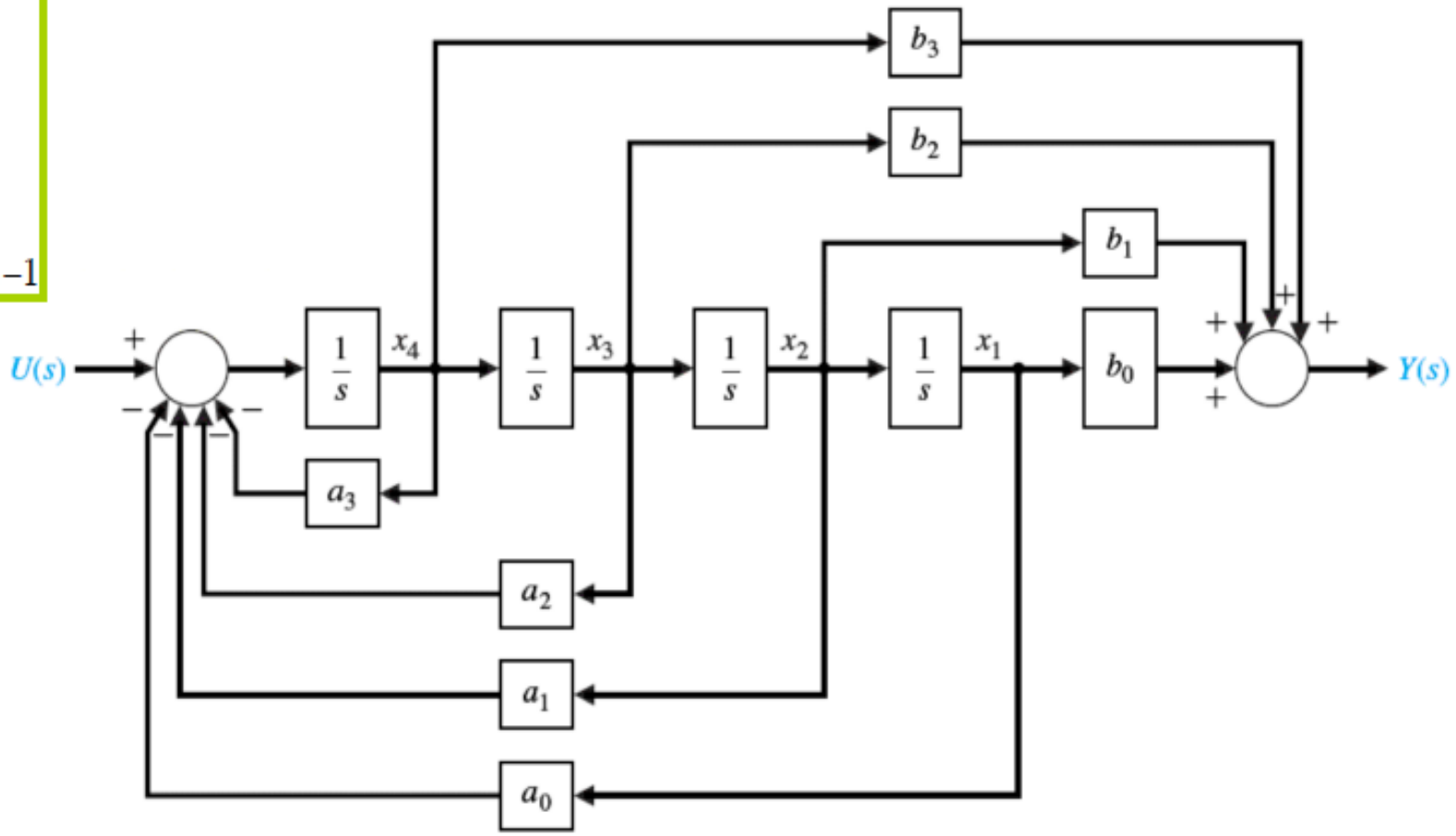
$$A = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -a_0 & -a_1 & -a_2 & \cdots & -a_{n-1} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}$$

$$C = \begin{bmatrix} b_0 & b_1 & b_2 & \cdots & b_{n-1} \end{bmatrix}, \quad D = 0$$

State-Space Block Diagram
Control Canonical Form

$$x_1 = z$$
$$x_2 = \dot{x}_1$$
$$x_3 = \dot{x}_2$$
$$\vdots$$
$$x_n = \dot{x}_{n-1}$$

$$\frac{Y(s)}{U(s)} = \frac{b_{n-1}s^{n-1} + \cdots + b_1 s + b_0}{s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0}$$



State-Space Block Diagram
Observer Canonical Form

$$\frac{Y(s)}{U(s)} = \frac{b_{n-1}s^{n-1} + \cdots + b_1 s + b_0}{s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0}$$

Let $y(t) = x_1(t)$, then

$$(s^n + a_{n-1}s^{n-1} + \ldots + a_1 s + a_0)X_n(s)$$
$$= (b_{n-1}s^{n-1} + \ldots + b_1 s + b_0)U(s)$$
$$- s^n X_n(s) + s^{n-1}(-a_{n-1}X_n(s) + b_{n-1}U(s)) +$$
$$\ldots + s(-a_1 X_n(s) + b_1 U(s)) + (-a_0 X_n(s) + b_0 U(s))$$
$$= 0$$

$sX_1(s)$

$s^2X_2(s)$

$s^nX_n(s)$

State-Space Block Diagram
Observer Canonical Form

State-Space Block Diagram
Observer Canonical Form

$$\frac{Y(s)}{U(s)} = \frac{b_{n-1}s^{n-1} + \cdots + b_1 s + b_0}{s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0}$$

$$A = \begin{bmatrix} 0 & 0 & \cdots & 0 & -a_0 \\ 1 & 0 & \cdots & 0 & -a_1 \\ 0 & 1 & \cdots & 0 & -a_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & a_{n-1} \end{bmatrix}, \quad B = \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ \vdots \\ b_{n-1} \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \end{bmatrix}, \quad D = 0$$

State-Space Block Diagram

Example 1
Given the transfer function

$$G(s) = \frac{s+2}{s^2+4s+3}$$

1. Find the state-space description in control canonical form.

2. Find the state-space description in observer canonical form.

State-Space Block Diagram

Example 1

- **1. Control canonical form**

$$A = \begin{bmatrix} 0 & 1 \\ -3 & -4 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix},$$
$$C = \begin{bmatrix} 2 & 1 \end{bmatrix}, \quad D = 0$$

- Results in the state-space realization

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -3 & -4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$
$$y = \begin{bmatrix} 2 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + 0 \cdot u$$

State-Space Block Diagram

Example 1

- **2. Observer canonical form**

$$A = \begin{bmatrix} 0 & -3 \\ 1 & -4 \end{bmatrix}, \quad B = \begin{bmatrix} 2 \\ 1 \end{bmatrix},$$
$$C = \begin{bmatrix} 0 & 1 \end{bmatrix}, \qquad D = 0$$

- Results in the state-space realization

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} 0 & -3 \\ 1 & -4 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \begin{bmatrix} 2 \\ 1 \end{bmatrix} u$$
$$y = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + 0 \cdot u$$

3.5 Converting a Transfer Function to State-Space

Differential equation:

$$\frac{d^n y}{dt^n} + a_{n-1}\frac{d^{n-1} y}{dt^{n-1}} + \ldots + a_1\frac{dy}{dt} + a_0 y = b_0 u$$

Phase-variable choice:

$$x_1 = y \qquad \dot{x}_1 = \frac{dy}{dt} \qquad \dot{x}_1 = x_2$$

$$x_2 = \frac{dy}{dt} \qquad \dot{x}_2 = \frac{d^2 y}{dt^2} \qquad \dot{x}_2 = x_3$$

$$x_3 = \frac{d^2 y}{dt^2} \rightarrow \dot{x}_3 = \frac{d^3 y}{dt^3} \rightarrow \dot{x}_3 = x_4$$

M M M

$$x_n = \frac{d^{n-1} y}{dt^{n-1}} \qquad \dot{x}_n = \frac{d^n y}{dt^n} \qquad \dot{x}_n = -a_0 x_1 - a_1 x_2 - a_2 x_3 - \ldots - a_{n-1} x_n + b_0 u$$

3.5 Converting a Transfer Function to State-Space

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \mathrm{M} \\ \dot{x}_{n-1} \\ \dot{x}_n \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & \mathrm{K} & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & \mathrm{K} & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & \mathrm{K} & 0 \\ \mathrm{M} & & & & & & \mathrm{K} & \\ 0 & 0 & 0 & 0 & 0 & 0 & \mathrm{K} & 1 \\ -a_0 & -a_1 & -a_2 & -a_3 & -a_4 & -a_5 & \mathrm{K} & -a_{n-1} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \mathrm{M} \\ x_{n-1} \\ x_n \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \mathrm{M} \\ 0 \\ b_0 \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 & 0 & \mathrm{L} & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \mathrm{M} \\ x_{n-1} \\ x_n \end{bmatrix}$$

3.5 Converting a Transfer Function to State-Space

Example 3.4

$$\frac{C(s)}{R(s)} = \frac{24}{\left(s^3 + 9s^2 + 26s + 24\right)}$$

$$\left(s^3 + 9s^2 + 26s + 24\right)C(s) = 24R(s)$$

$$\dddot{c} + 9\ddot{c} + 26\dot{c} + 24c = 24r$$

State variables

$$x_1 = c \qquad \dot{x}_1 = x_2$$

$$x_3 = \ddot{c} \rightarrow \dot{x}_3 = x_3$$

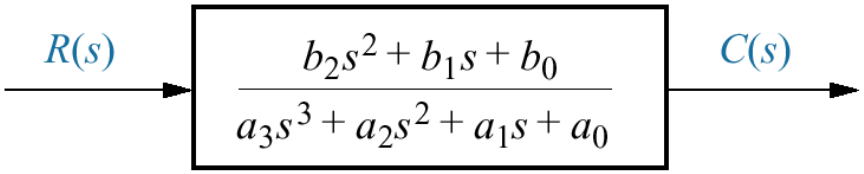
$$x_3 = \dddot{c} \qquad \dot{x}_3 = -24x_1 - 26x_2 - 9x_3 + 24r$$

$$y = c = x_1$$

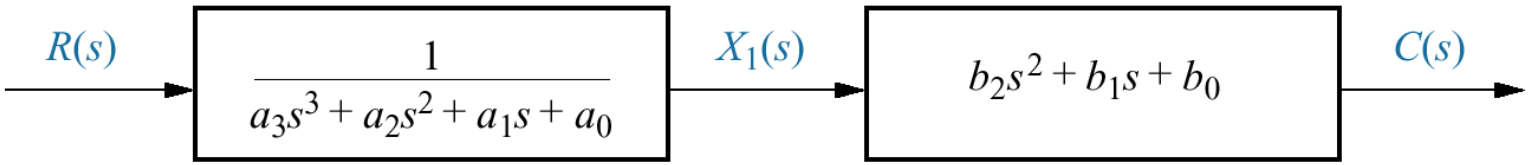
$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -24 & -26 & -9 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 24 \end{bmatrix} \qquad \text{and} \qquad y = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

3.5 Converting a Transfer Function to State-Space

Transfer function with polynomial numerator



(a)



Internal variables:
$X_2(s), X_3(s)$

(b)

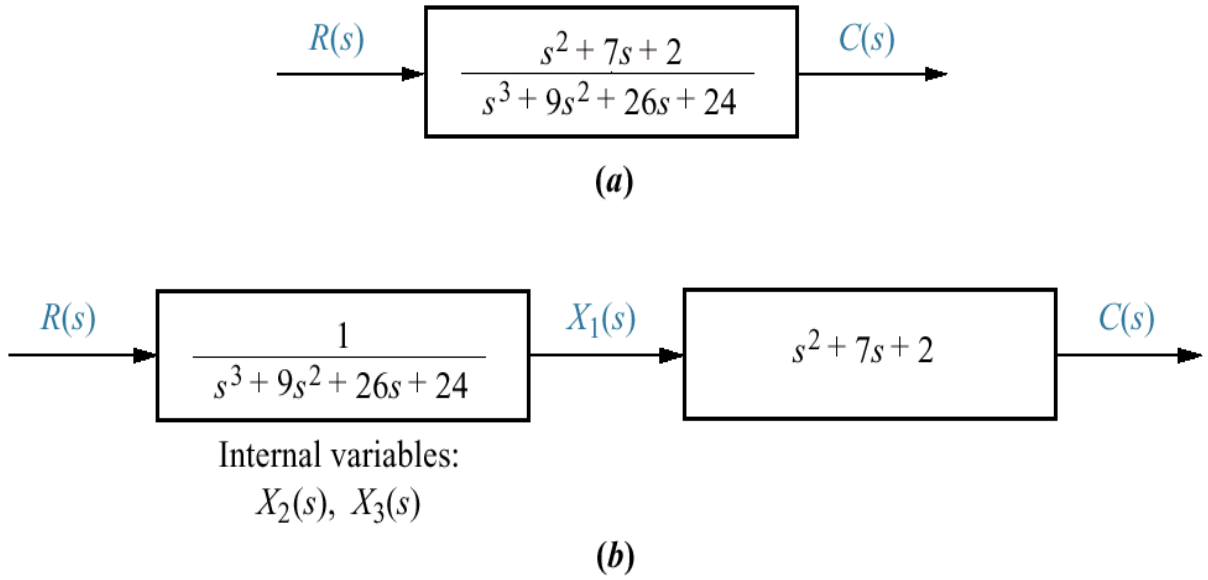
$$Y(s) = C(s) = (b_2s^2 + b_1s + b_0)x_1(s)$$

$$y(t) = b_2\frac{d^2x_1}{dt^2} + b_1\frac{dx_1}{dt} + b_0x_1$$

$$= b_0x_1 + b_1x_2 + b_2x_3$$

3.5 Converting a Transfer Function to State-Space

Examlpe 3.5



$$C(s) = \left(s^2 + 7s + 2\right) X_1(s)$$

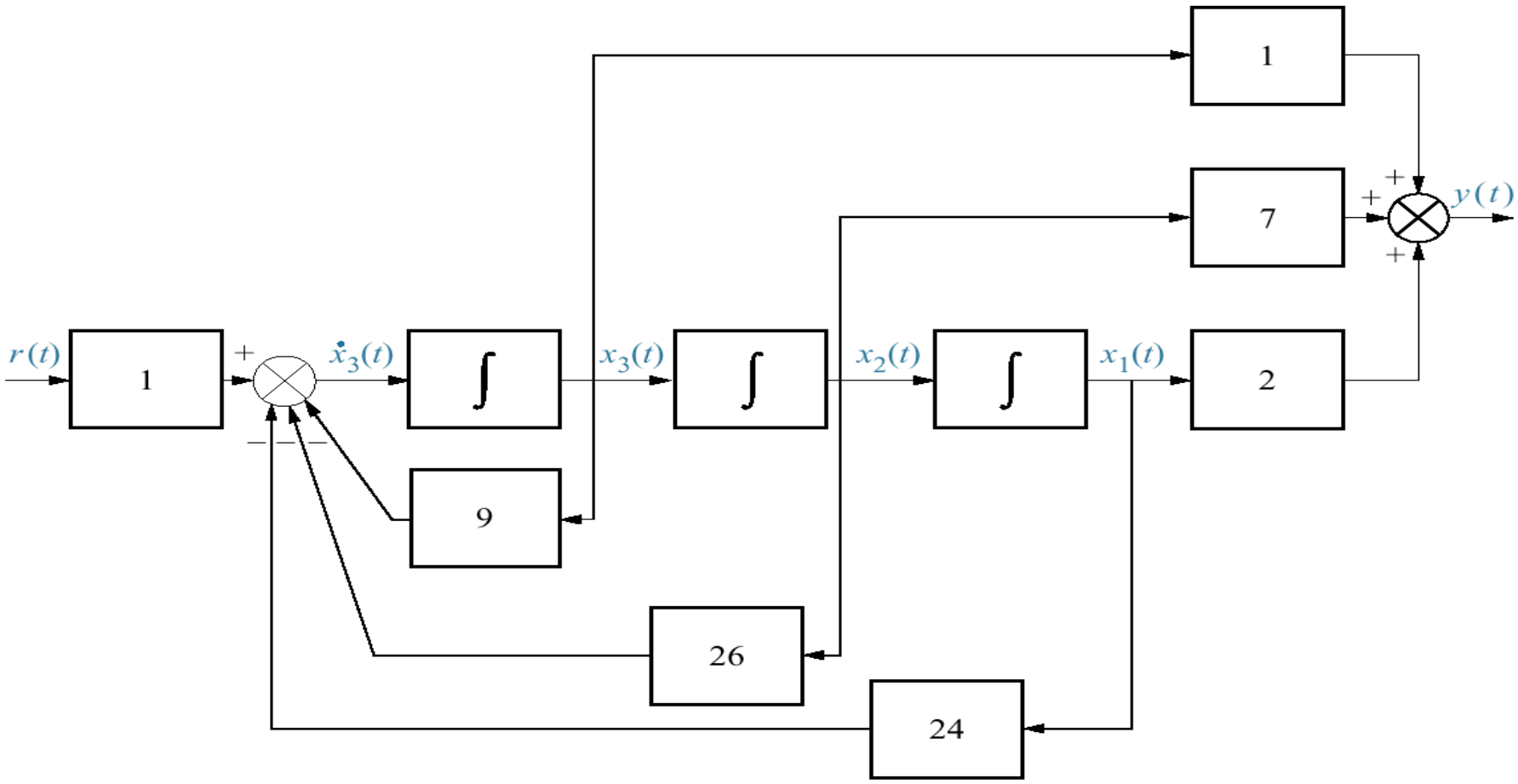
$$c = \ddot{x}_1 + 7\dot{x}_1 + 2x_1$$

$$y = 2x_1 + 7x_2 + x_3$$

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -24 & -26 & -9 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u$$

$$y = \begin{bmatrix} 2 & 7 & 1 \end{bmatrix} x + \begin{bmatrix} 0 \end{bmatrix} u$$

3.5 Converting a Transfer Function to State-Space

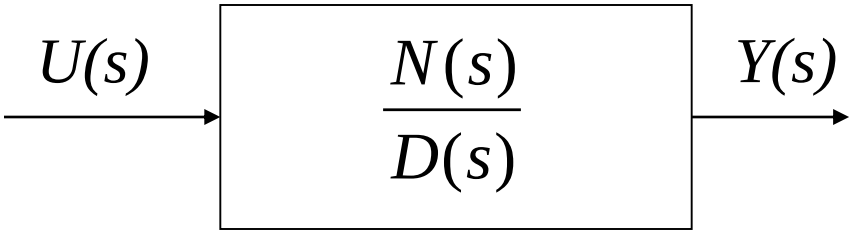


(c)

3.5 Converting a Transfer Function to State-Space

$$Y(s) = \frac{N(s)}{D(s)} U(s)$$

$$D(s)Y(s) = N(s)U(s)$$



Denominator: differentiates the output and integrates the input.

Numerator: differentiates the input and integrates the output.

3.6 Converting from State-Space to Transfer Function

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u$$
$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u}$$

Apply Laplace transform

$$s\mathbf{X}(s) = \mathbf{A}\mathbf{X}(s) + \mathbf{B}\mathbf{U}(s)$$
$$\mathbf{Y}(s) = \mathbf{C}\mathbf{X}(s) + \mathbf{D}\mathbf{U}(s)$$

$$(s\mathbf{I} - \mathbf{A})\mathbf{X}(s) = \mathbf{B}\mathbf{U}(s)$$
$$\mathbf{X}(s) = (s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}\mathbf{U}(s)$$

$$\mathbf{Y}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}\mathbf{U}(s) + \mathbf{D}\mathbf{U}(s)$$
$$= \left[\mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D}\right]\mathbf{U}(s)$$
$$T(s) = \frac{Y(s)}{U(s)} = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D}$$

3.6 Converting from State-Space to Transfer Function

Example 3.6

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -1 & -2 & -3 \end{bmatrix} x + \begin{bmatrix} 10 \\ 0 \\ 0 \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \end{bmatrix} u$$

$$(s\mathbf{I} - \mathbf{A}) = \begin{bmatrix} s & 0 & 0 \\ 0 & s & 0 \\ 0 & 0 & s \end{bmatrix} - \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -1 & -2 & -3 \end{bmatrix} = \begin{bmatrix} s & -1 & 0 \\ 0 & s & -1 \\ 1 & 2 & s+3 \end{bmatrix}$$

$$(s\mathbf{I} - \mathbf{A})^{-1} = \frac{\text{adj}(s\mathbf{I} - \mathbf{A})}{\det(s\mathbf{I} - \mathbf{A})} = \frac{\begin{bmatrix} s^2 + 3s + 2 & s+3 & 1 \\ -1 & s(s+3) & s \\ -s & -(2s+1) & s^2 \end{bmatrix}}{s^3 + 3s^2 + 2s + 1}$$

3.6 Converting from State-Space to Transfer Function

$$T(s)=\frac{\begin{bmatrix}1 & 0 & 0\end{bmatrix}\begin{bmatrix}s^2+3s+2 & s+3 & 1\\ -1 & s(s+3) & s\\ -s & -(2s+1) & s^2\end{bmatrix}\begin{bmatrix}10\\0\\0\end{bmatrix}}{s^3+3s^2+2s+1}+\begin{bmatrix}0\end{bmatrix}$$

$$T(s)=\frac{10\left(s^2+3s+2\right)}{s^3+3s^2+2s+1}$$

3.6 Converting from State-Space to Transfer Function

Matrix Inverse

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \quad \Rightarrow \quad \mathbf{A}^{-1} = \frac{adj\left(\mathbf{A}\right)}{\det\left(\mathbf{A}\right)}$$

$$adj\left(\mathbf{A}\right) = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix}^{T}, \quad c_{ij} = \left(-1\right)^{(i+j)} M_{ij}$$

Minor $M_{ij} = \left|\mathbf{A}\right|$ after eliminating the i^{th} row and the j^{th} column.

$$\det\left(\mathbf{A}\right) = \sum_{k=1}^{n} a_{ik} c_{ik}$$

3.6 Converting from State-Space to Transfer Function

Example

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 3 \\ -1 & 4 & 5 \\ 6 & 8 & 7 \end{bmatrix}$$

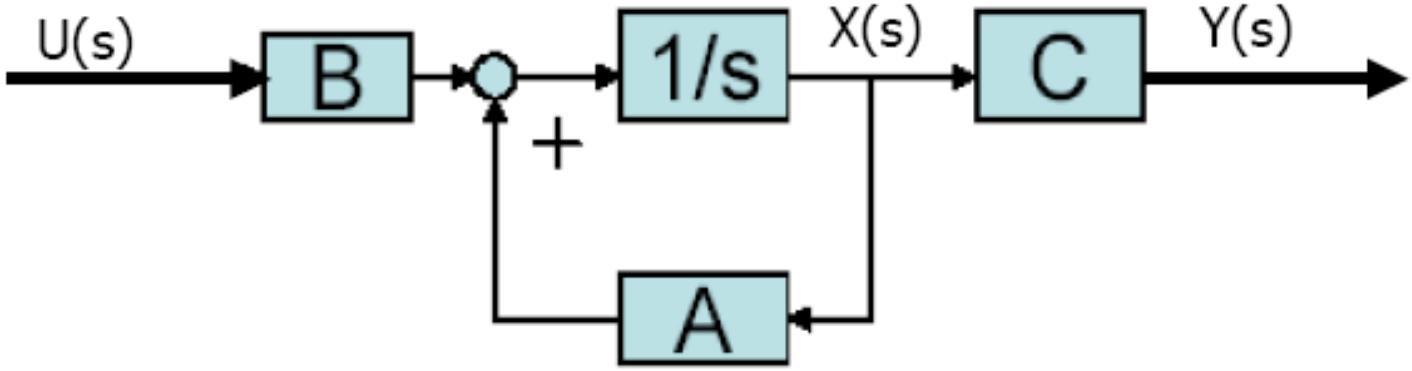
$$\text{adj}(\mathbf{A}) = \begin{bmatrix} \begin{vmatrix} 4 & 5 \\ 8 & 7 \end{vmatrix} & -\begin{vmatrix} -1 & 5 \\ 6 & 7 \end{vmatrix} & \begin{vmatrix} -1 & 4 \\ 6 & 8 \end{vmatrix} \\ -\begin{vmatrix} 2 & 3 \\ 8 & 7 \end{vmatrix} & \begin{vmatrix} 1 & 3 \\ 6 & 7 \end{vmatrix} & -\begin{vmatrix} 1 & 2 \\ 6 & 8 \end{vmatrix} \\ \begin{vmatrix} 2 & 3 \\ 4 & 5 \end{vmatrix} & -\begin{vmatrix} 1 & 3 \\ -1 & 5 \end{vmatrix} & \begin{vmatrix} 1 & 2 \\ -1 & 4 \end{vmatrix} \end{bmatrix}^T = \begin{bmatrix} -12 & 10 & -2 \\ 37 & -11 & -8 \\ -32 & 4 & 6 \end{bmatrix}$$

$$\det(\mathbf{A}) = 1(-12) + 2(37) + 3(-32) = -34$$

State-Space Block Diagram

With zero " initial state : $s\,X(s) = A\,X(s) + B\,U(s)$

With D = 0 : $Y(s) = C\,X(s)$



T.F. to State-Space

1. For a given transfer function, there is no unique state space realization

2. Engineering dictates the use of a realization of least order, a minimal realization

3. A minimal realization is both **controllable** and **observable**.

The State Transition Matrix

1 Previously, we found that

$$\dot{x}(t) = A\,x(t) + B\,u(t)$$

Thus, $X(s) = (sI - A)^{-1}\,x(0) + (sI - A)^{-1}\,B\,U(s)$

2 Now consider the homogenous (i.e. zero-input) case: $\dot{x} = Ax$

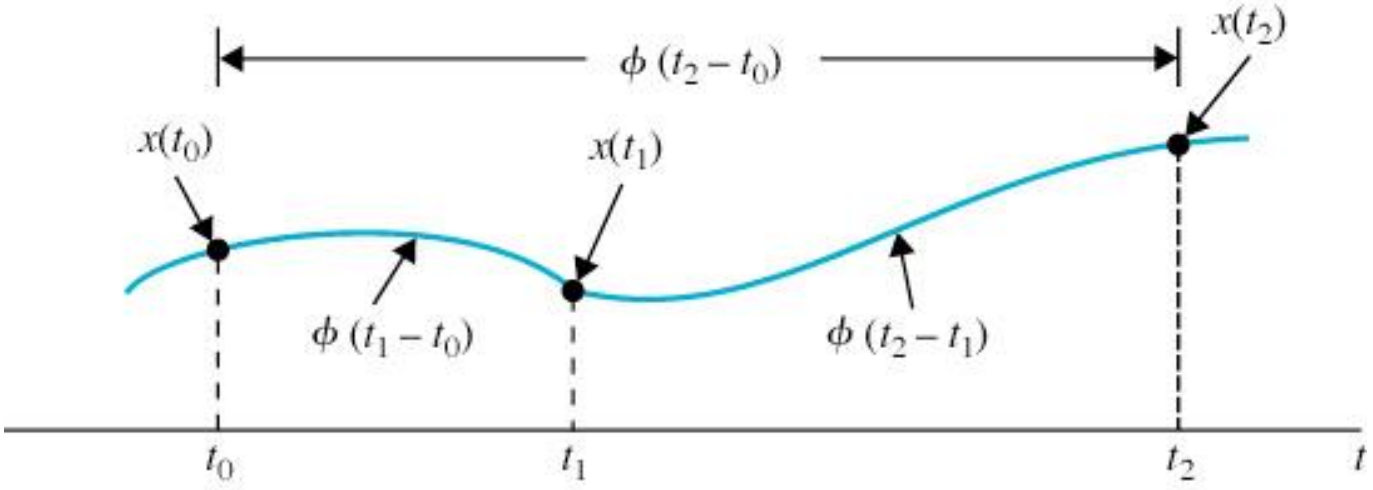
3 The solution to this equation represents the evolution of the system's free response to non-zero initial conditions:

$$x(t) = \Phi(t)x(0)$$

State transition matrix

The State Transition Matrix

Given an initial value, the state transition matrix predicts the state at any other time



How do we compute this matrix?

The State Transition Matrix

Successive differentiation provides

$$\begin{aligned} \dot{x} &= Ax \\ \ddot{x} &= A\dot{x} = A^2x \\ &\vdots \\ x^{(k)} &= A^kx \end{aligned}$$

Evaluate at t = 0:

$$x^{(k)}(0) = A^kx(0)$$

The State Transition Matrix

- Consider the homogenous (i.e. zero-input) dynamics:

$$\dot{x} = Ax$$

- A Taylor's series approximation taken about *t=0* provides the solution

$$x(t) \quad = \quad e^{At}x(0) = \Phi(t)x(0)$$

- In which the **matrix exponential** is defined as

$$e^{At} = I + At + A^2\frac{t^k}{2!} + \ldots + A^k\frac{t^k}{k!} + \ldots = \sum_{k=1}^{\infty} A^k\frac{t^k}{k!}$$

The State Transition Matrix

Matrix exponential properties

$$
\begin{aligned}
e^{A\cdot 0} &= I \\
e^{A(t_1+t_2)} &= e^{At_1}e^{At_2} = e^{At_2}e^{At_1} \\
\left(e^{At}\right)^{-1} &= e^{-At} \\
e^{A^T t} &= \left(e^{At}\right)^T \\
Ae^{At} &= e^{At}A \\
\frac{d}{dt}e^{At} &= Ae^{At}
\end{aligned}
$$

The State Transition Matrix

- Instead of solving in the time domain, consider

$$\dot{x}(t) = Ax(t)$$
$$sX(s) - x(0) = AX(s)$$
$$(sI - A)X(s) = x(0)$$
$$X(s) = (sI - A)^{-1}x(0) = \Phi(s)x(0)$$

- In the Laplace domain, the state transition matrix is

$$\Phi(s) = (sI - A)^{-1}$$

- therefore

$$\Phi(t) = L^{-1}\left(\Phi(s)\right) = L^{-1}\left((sI - A)^{-1}\right)$$

The State Transition Matrix

- Consider the inhomogeneous (forced) case:

$$\dot{x} = Ax + Bu$$

$$e^{-At}(\dot{x} - Ax) = e^{-At}Bu$$

$$\frac{d}{dt}(e^{-At}x) = e^{-At}Bu$$

$$\int_0^t \frac{d}{d\tau}(e^{-A\tau}x)d\tau = \int_0^t e^{-A\tau}Bu(\tau)d\tau$$

$$e^{-At}x(t) - e^{-A0}x(0) = \int_0^t e^{-A\tau}Bu(\tau)d\tau$$

$$x(t) = e^{At}x(0) + \int_0^t e^{-A(t-\tau)}Bu(\tau)d\tau$$

$$= \Phi(t)x(0) + \int_0^t \Phi(t-\tau)Bu(\tau)d\tau$$

$$\Phi(t) = e^{At}$$

The State Transition Matrix

- Take the Laplace transform

$$\dot{x} = Ax + Bu$$

$$sX(s) - x(0) = AX(s) + BU(s)$$

$$X(s) = (sI - A)^{-1}x(0) + (sI - A)^{-1}BU(s)$$

$$X(s) = \Phi(s)x(0) + \Phi(s)BU(s), \quad \Phi(s) = (sI - A)^{-1}$$

- Matches the time-domain solution

$$x(t) = \Phi(t)x(0) + \int_0^t \Phi(t-\tau)Bu(\tau)d\tau$$

$$\Phi(t) = e^{At}$$

Natural response

Forced response

The State Transition Matrix

Note that the system response has two components:

- Natural response – "zero input response" due to initial conditions
- Forced response – "zero state response" due to input

Overall response is the sum of the two

$$x(t) = \Phi(t)x(0) + \int_0^t \Phi(t-\tau)Bu(\tau)d\tau$$

$$\Phi(t) = e^{At}$$

Natural response

Forced response

The State Transition Matrix

- Given $A = \begin{bmatrix} 0 & -2 \\ 1 & -3 \end{bmatrix}$,
- The state transition matrix is

$$\begin{aligned} \Phi(s) &= (sI - A)^{-1} \\ &= \begin{bmatrix} s & 2 \\ -1 & s+3 \end{bmatrix}^{-1} \\ &= \frac{1}{\Delta(s)} \begin{bmatrix} s+3 & -2 \\ 1 & s \end{bmatrix}, \text{ where} \\ \Delta(s) &= s^2 + 3s + 2 = (s+1)(s+2) \end{aligned}$$

- The time-domain state transition matrix can be obtained using the inverse Laplace transform

The State Transition Matrix

- Find $\Phi(t) = \mathcal{L}^{-1}(\Phi(s))$

$$= \frac{1}{(s+1)(s+3)} \begin{bmatrix} s+3 & -2 \\ 1 & s \end{bmatrix}$$

- And using known Inverse Laplace Transforms (Table D.1, Dorf and Bishop),

$\frac{1}{(s+a)(s+b)}$	$\frac{1}{(b-a)}(e^{-at} - e^{-bt})$
$\frac{s+\alpha}{(s+a)(s+b)}$	$\frac{1}{(b-a)}[(\alpha - a)e^{-at} - (\alpha - b)e^{-bt}]$

$$\Phi(t) = \begin{bmatrix} 2e^{-t} - e^{-2t} & -2e^{-t} + 2e^{-2t} \\ e^{-t} - e^{-2t} & -e^{-t} + 2e^{-2t} \end{bmatrix}$$

The State Transition Matrix

- State transition matrix (time domain)

$$\Phi(t) = \begin{bmatrix} 2e^{-t} - e^{-2t} & -2e^{-t} + 2e^{-2t} \\ e^{-t} - e^{-2t} & -e^{-t} + 2e^{-2t} \end{bmatrix}$$

- With initial conditions $x_0 = [1\ 1]^T$, the free (unforced) response is

$$\begin{aligned} x(t) &= \Phi(t)x(0) \\ &= \begin{bmatrix} e^{-2t} \\ e^{-2t} \end{bmatrix} \end{aligned}$$

Linearization

· Need Linear-Time Invariant Models for Applying Modern Control Theory.

· Most Practical Control Problems are Nonlinear.

· Taylor Series Linearization Provides a Direct Way to Derive Linear Approximation to a Nonlinear System.

2.10 Nonlinearities

A linear system has two properties:

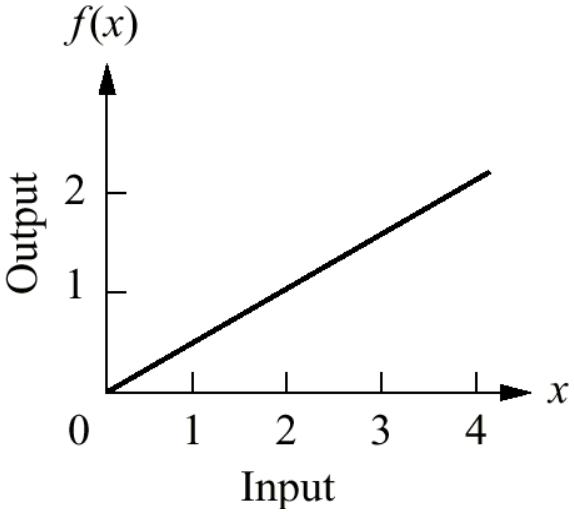
- Superposition:

$$\frac{\begin{array}{c} r_1(t) \rightarrow c_1(t) \\ r_2(t) \rightarrow c_2(t) \end{array}}{r_1(t) + r_2(t) \rightarrow c_1(t) + c_2(t)}$$

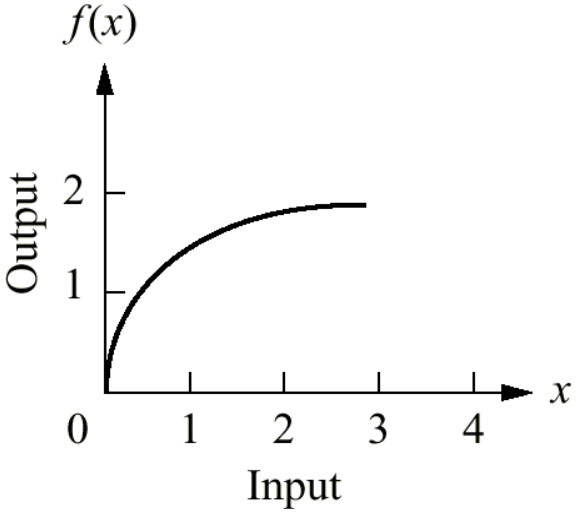
- Homogeniety:

$$\frac{r_1(t) \rightarrow c_1(t)}{A \cdot r_1(t) \rightarrow A \cdot c_1(t)}$$

Linear-/nonlinear System



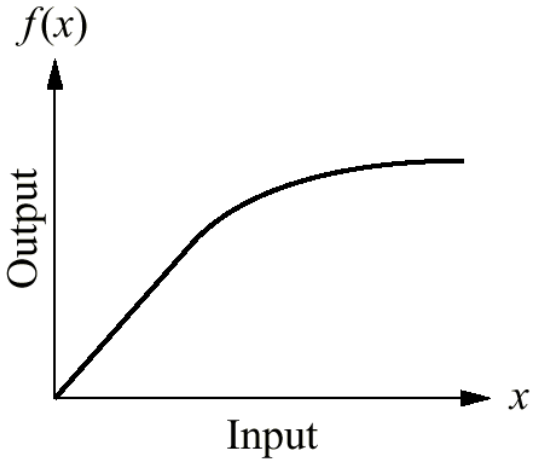
a. Linear system.



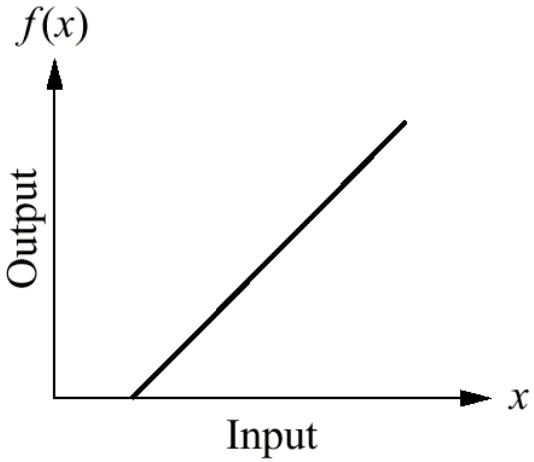
b. Nonlinear system

Some physical nonlinearity

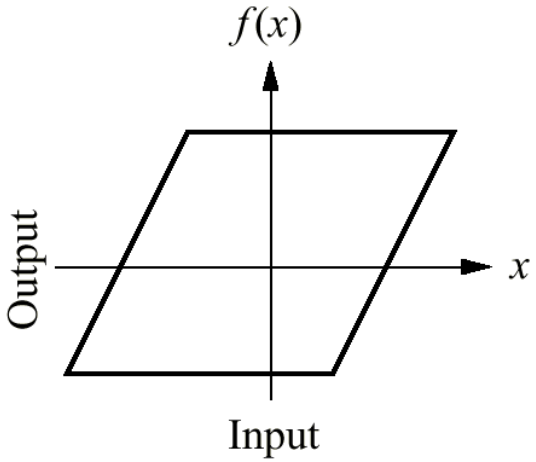
Amplifier saturation



Motor dead zone



Backlash in gears



Linearization

· A Given Nonlinear System:

$$\dot{\mathbf{X}} = \mathbf{F}(\mathbf{X}, \mathbf{U})$$
$$\mathbf{Y} = \mathbf{G}(\mathbf{X}, \mathbf{U})$$

$$\mathbf{X} \in \Re^n, \mathbf{U} \in \Re^m, \mathbf{Y} \in \Re^r$$

Define a Nominal Condition:

$$\underline{\mathbf{X}}, \quad \underline{\mathbf{U}}, \quad \underline{\mathbf{Y}}$$

Such that

$$\mathbf{X} = \underline{\mathbf{X}} + \delta\mathbf{X}, \quad \mathbf{U} = \underline{\mathbf{U}} + \delta\mathbf{U}, \quad \mathbf{Y} = \underline{\mathbf{Y}} + \delta\mathbf{Y}$$

and $\dot{\mathbf{X}} = \dot{\underline{\mathbf{X}}} + \delta\dot{\mathbf{X}}$

Linearization

Linearization Around the Stationary State

Taylor series expansion is to calculate $\Delta\dot{x} = f(\Delta x, \Delta u)$

$$
\begin{aligned}
f(\Delta x, \Delta u) &= f(\overline{x}, \overline{u}) + \frac{\partial f}{\partial x}\Big|_{x=\overline{x}, u=\overline{u}} \Delta x + \frac{\partial f}{\partial u}\Big|_{x=\overline{x}, u=\overline{u}} \Delta u + \ldots, \\
\Delta\dot{x}_i &= \sum_{k=1}^{n} \left(\frac{\partial f_i}{\partial x_k}\right)\Big|_{x=\overline{x}, u=\overline{u}} \Delta x_k + \sum_{j=1}^{m} \left(\frac{\partial f_i}{\partial u_j}\right)\Big|_{x=\overline{x}, u=\overline{u}} \Delta u_j, \\
\Delta\dot{x} &= \underbrace{A}_{\text{Jacobian matrix}} \Delta x + \underbrace{B}_{\text{input matrix}} \Delta u.
\end{aligned}
$$

Linearization

For the nonlinear system $\dot{\mathbf{X}} = \mathbf{F}(\mathbf{X},\mathbf{U})$, $\mathbf{Y} = \mathbf{G}(\mathbf{X},\mathbf{U})$

Where A, B, C, D are Matrices of Partial Derivatives (Jacobian Matrix) Evaluated at the Nominal Values:

$$\mathbf{A} = \left.\frac{\partial \mathbf{F}}{\partial \mathbf{X}}\right|_{\substack{\mathbf{X}=\underline{\mathbf{X}} \\ \mathbf{U}=\underline{\mathbf{U}}}} \qquad \mathbf{B} = \left.\frac{\partial \mathbf{F}}{\partial \mathbf{U}}\right|_{\substack{\mathbf{X}=\underline{\mathbf{X}} \\ \mathbf{U}=\underline{\mathbf{U}}}}$$

$$\mathbf{C} = \left.\frac{\partial \mathbf{G}}{\partial \mathbf{X}}\right|_{\substack{\mathbf{X}=\underline{\mathbf{X}} \\ \mathbf{U}=\underline{\mathbf{U}}}} \qquad \mathbf{D} = \left.\frac{\partial \mathbf{G}}{\partial \mathbf{U}}\right|_{\substack{\mathbf{X}=\underline{\mathbf{X}} \\ \mathbf{U}=\underline{\mathbf{U}}}}$$

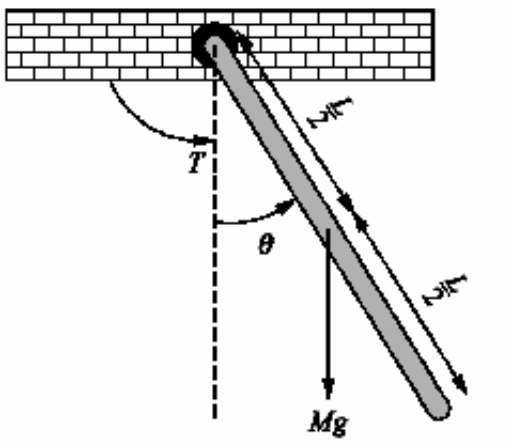
$$A = \begin{bmatrix} \dfrac{\partial f_1}{\partial x_1} & \dfrac{\partial f_1}{\partial x_2} & \cdots & \dfrac{\partial f_1}{\partial x_n} \\ \vdots & & & \\ \dfrac{\partial f_n}{\partial x_1} & \dfrac{\partial f_n}{\partial x_2} & \cdots & \dfrac{\partial f_n}{\partial x_n} \end{bmatrix}_{\overline{x},\overline{u}}$$

$$B = \begin{bmatrix} \dfrac{\partial f_1}{\partial u_1} & \dfrac{\partial f_1}{\partial u_2} & \cdots & \dfrac{\partial f_1}{\partial u_n} \\ \vdots & & & \\ \dfrac{\partial f_n}{\partial u_1} & \dfrac{\partial f_n}{\partial u_2} & \cdots & \dfrac{\partial f_n}{\partial u_n} \end{bmatrix}_{\overline{x},\overline{u}}.$$

3.7 Linearization

State-sapce representation allows for nonlinearities

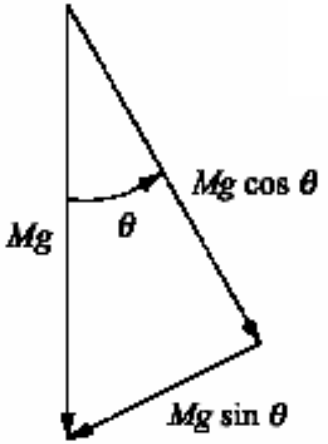
$$T = J\frac{d^2\theta}{dt^2} + \frac{MgL}{2}\sin(\theta)$$



Assume $u = T$

$y = \theta$

To find the transfer function,we must linearize the equation.



State-space representation

$$x_1 = \theta$$

$$x_2 = \frac{d\theta}{dt}$$

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = \frac{-MgL}{2J}\sin(x_1) + \frac{T}{J}$$

Linearization

Operating point: $x_1 = 0,\ x_2 = 0\ (\theta = \dot{\theta} = 0)$

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = \dot{x}_2(0) + \left.\frac{d\dot{x}_2}{dx_1}\right|_{x_1=0} (x_1 - 0)$$

$$= \frac{T}{J} - \frac{MgL}{2J}\cos(0)(x_L)$$

$$= \frac{T}{J} - \frac{MgL}{2J}(x_1)$$

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{MgL}{2J} & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{J} \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Linearization

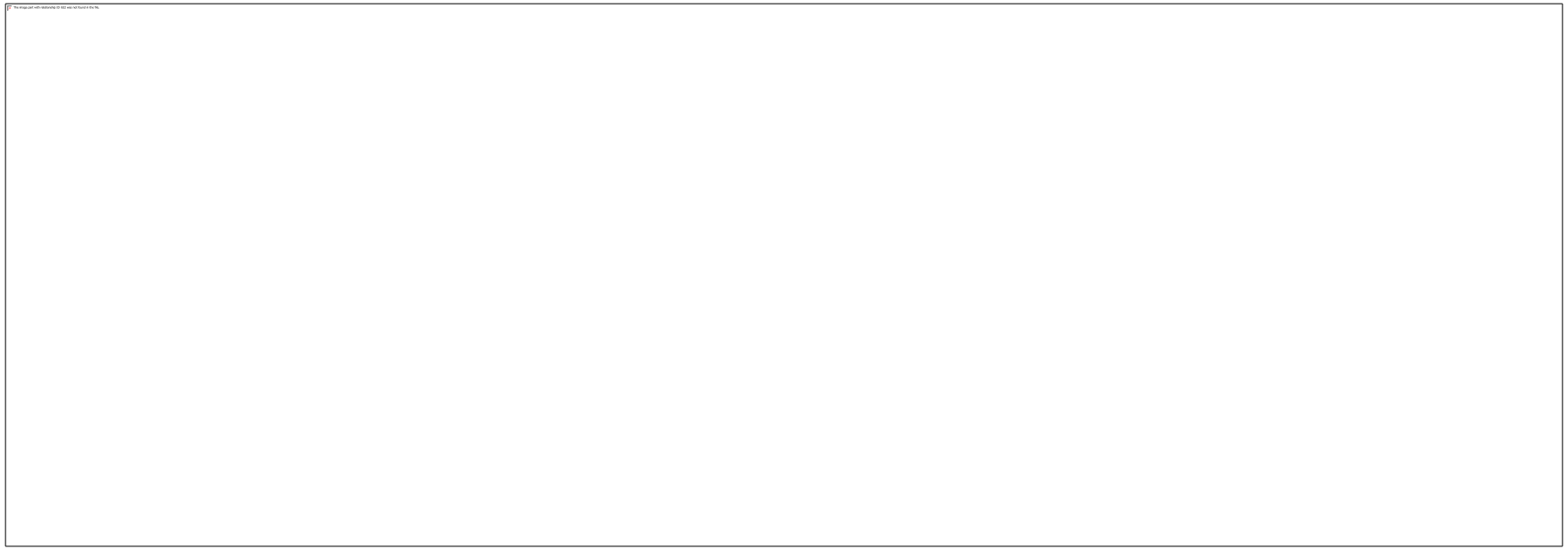
Given the model of Watt's governor

$$\dot{x}_1 = x_2,$$
$$\dot{x}_2 = \frac{1}{2} N^2 x_3^2 \sin 2x_1 - g \sin x_1 - \frac{b}{m} x_2,$$
$$\dot{x}_3 = \frac{\kappa}{I} \cos x_1 - \frac{\tau}{I}.$$

Linearize the above model of an equilibrium state of the form

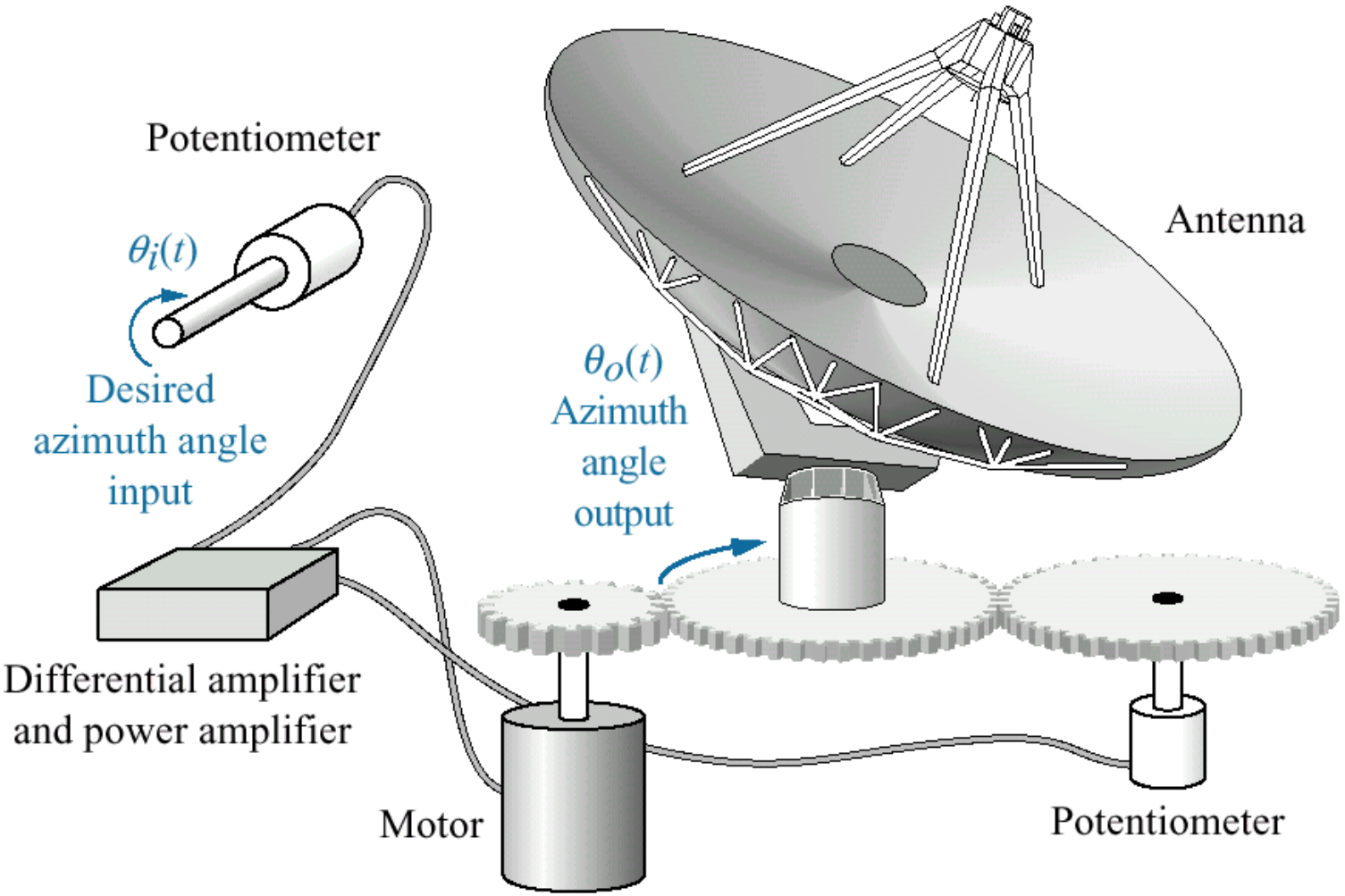
$$\boldsymbol{x}_e = [x_{1e} \quad 0 \quad x_{3e}]^T.$$

Linearization



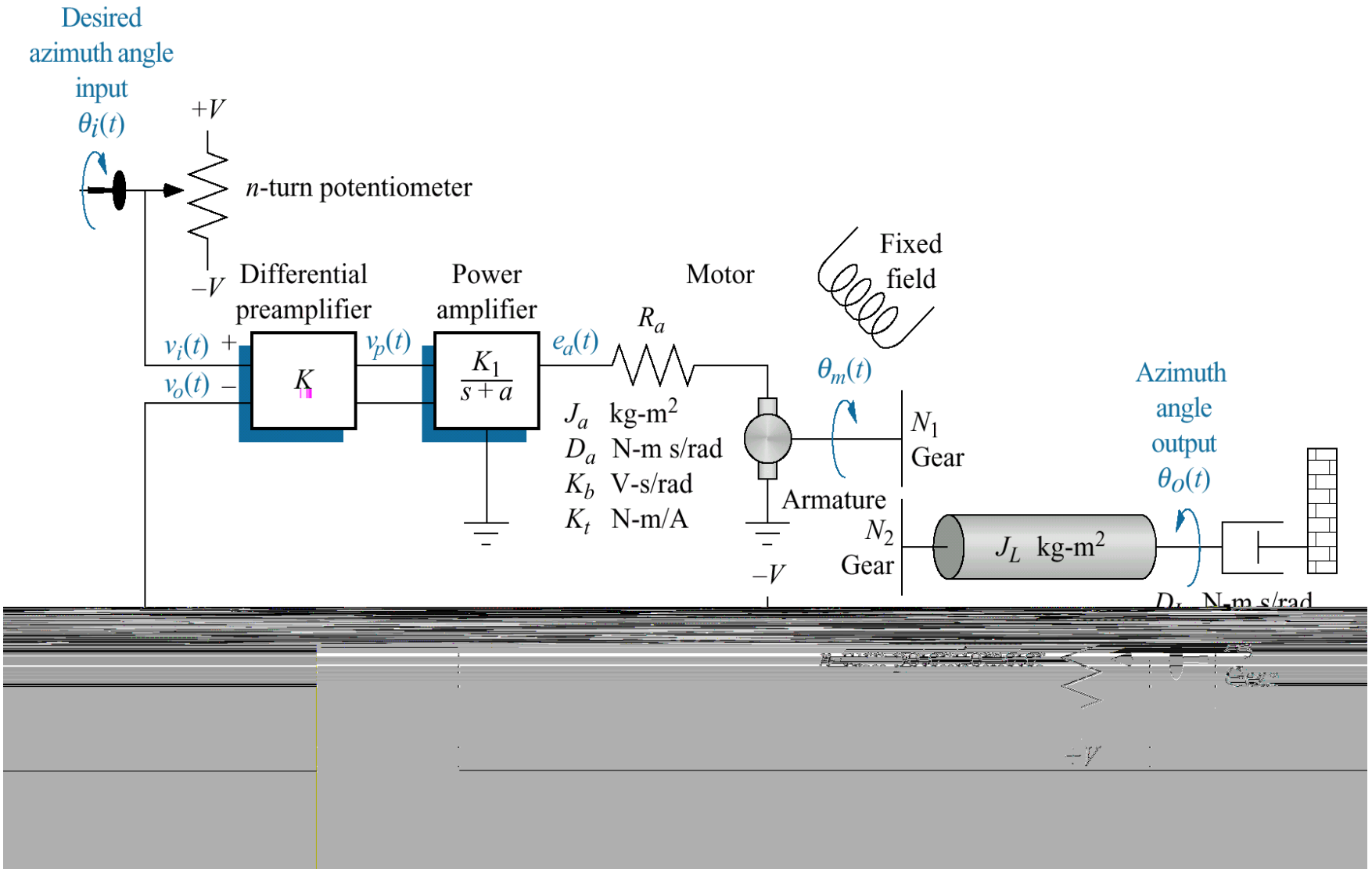
1. Antenna Azimuth Position Control System

Layout



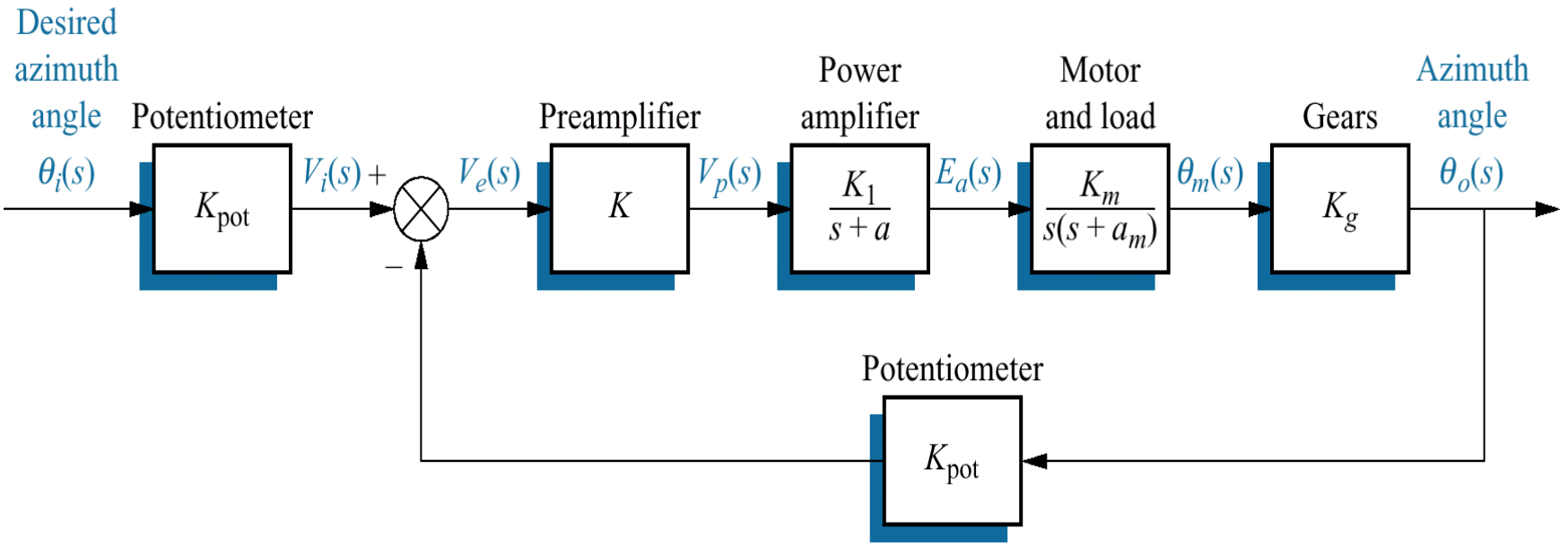
2. Antenna Azimuth Position Control System

Schematic



3. Antenna Azimuth Position Control System

Block Diagram



4. Antenna Azimuth Position Control System

Schematic Parameters

Parameter	Configuration 1	Configuration 2	Configuration 3
V	10	10	10
n	10	1	1
K	—	—	—
K_1	100	150	100
a	100	150	100
R_a	8	5	5
L	0.02	0.05	0.05

5. Antenna Azimuth Position Control System

Block Diagram Parameters

Parameter	Configuration 1	Configuration 2	Configuration 3
K_{pot}	0.318		
K	—		
K_1	100		
a	100		
K_m	2.083		
a_m	1.71		
K_g	0.1		

Power Amplifer

$$\frac{E_a(s)}{V_p(s)} = G(s) = \frac{100}{s+100}$$

$$(s+100)E_a(s) = 100V_p(s)$$

$$\frac{de_a}{dt} + 100e_a = 100v_p$$

$$\frac{de_a}{dt} = -100e_a + 100v_p$$

Output

$$y = e_a$$

$$\dot{x}_1 = -100x + 100v_p$$

$$y = x$$

Motor and load

$$e_a(t) = i_a(t) R_a + K_b(t) \frac{d\theta}{dt}$$

$$T_m(t) = K_t i_a(t) = J_m \frac{d^2\theta_m}{dt^2} + D_m \frac{d\theta_m}{d_t}$$

$$e_a(t) = \left(\frac{R_a J_m}{K_t} \right) \frac{d^2\theta_m}{dt^2} + \left(\frac{D_m R_a}{K_t} + K_b \right) \frac{d\theta_m}{dt}$$

State variables

$$x_1 = \theta_m \qquad x_2 = \frac{d\theta_m}{dt}$$

$$e_a(t) = \left(\frac{R_a J_m}{K_t} \right) \frac{dx_2}{dt} + \left(\frac{D_m R_a}{K_t} + K_b \right) x_2$$

$$\frac{dx_2}{dt} = -\frac{1}{J_m} \left(D_m + \frac{K_t K_b}{R_a} \right) x_2 + \left(\frac{K_t}{R_a J_m} \right) e_a(t)$$

$$\frac{dx_1}{dt} = x_2$$

Output $\theta_o(t)$

$$y = 0.1x_1$$

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & -\frac{1}{J_m}\left(D_m + \frac{K_t K_b}{R_a}\right) \end{bmatrix} x + \begin{bmatrix} 0 \\ \frac{K_t}{R_a J_m} \end{bmatrix} e_a(t)$$

$$y = \begin{bmatrix} 0.1 & 0 \end{bmatrix} x$$

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & -1.71 \end{bmatrix} x + \begin{bmatrix} 0 \\ 2.083 \end{bmatrix} e_a(t)$$

$$y = \begin{bmatrix} 0.1 & 0 \end{bmatrix} x$$

To combine the two subsystems:

1. Multiply the two transfer functions and then transfer the result into state-space representation, or
2. Define three state variables corresponding to previous states.

$$x_1 = e_a, x_2 = \theta_m, x_3 = \frac{d\theta_m}{dt}$$

$$\dot{x}_1 = -100x_1 + 100v_p(t)$$

$$y_1 = x_1$$

$$\dot{x}_2 = x_3$$

$$\dot{x}_3 = -1.7x_3 + 2.083e_a(t)$$

$$= -1.7x_3 + 2.083x_1$$

$$y_2 = 0.1x_2$$

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} -100 & 0 & 0 \\ 0 & 0 & 1 \\ 2.083 & 0 & -1.7 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 100 \\ 0 \\ 0 \end{bmatrix} v_p(t) \qquad y_2 = \begin{bmatrix} 0 & 0.1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

Linearly Independent

Two or more functions, equations, or vectors f_1, f_2, ..., which are not linearly dependent

$$a_1 f_1 + a_2 f_2 + \ldots + a_n f_n = 0$$

with a_1, a_2, ... constants which are not all zero are said to be linearly independent.

A set of n vectors $\mathbf{v}_1$, $\mathbf{v}_2$, ..., $\mathbf{v}_n$ is linearly independent iff the matrix rank of the matrix $\mathsf{m} = (\mathbf{v}_1\ \mathbf{v}_2\ \ldots\ \mathbf{v}_n)$ is n, in which case m is diagonalizable.